

Trabalho #2 – Renderização de Listas, Persistência (localStorage), Componentes e Rotas

Desenvolver um App utilizando o framework React JS contendo os seguintes requisitos:

- a) A página deve renderizar uma lista de dados, salvos (persistidos) em localStorage, contendo dados do projeto escolhido pelo aluno (nome, descrição, preço, foto, data ...)
- b) O usuário deve poder adicionar novos dados a esta lista. Exibir o item na listagem e salvar o novo registro em localStorage.
- c) O usuário deve poder interagir sobre a lista, a partir do clique em um botão associado a cada item e preencher/selecionar uma opção. Os dados desta interação devem ser salvos em localStorage.
- d) Organizar a aplicação em componentes (no mínimo, 3).
- e) O App deve conter rotas para exibir a página principal e, no mínimo, mais uma página – com recursos como, por exemplo, um formulário para filtrar os dados

Exemplos (* pode ser em duplas, cada dupla deve pensar uma aplicação diferente)

1. Uma **Agência de Turismo** disponibiliza uma lista de viagens para os seus clientes, salvos em localStorage. Novas viagens podem ser adicionadas pelo administrador. A interação do cliente pode ser para registrar interesse na viagem – contando +1 no atributo numInteressados (que deve começar zerado). Pode-se também utilizar 2 campos (nome e whats), declarados como vetores e acrescentar estes dados do cliente na viagem quando ele(a) indicar interesse.
2. Uma **floricultura** disponibiliza uma lista de arranjos para o Natal, por exemplo, salvos em localStorage. Novos arranjos podem ser adicionados pelo administrador. A interação dos clientes sobre os arranjos pode ser para indicar um Gostei (like) ou Não Gostei (dislike). Estes campos devem iniciar zerados no cadastro dos arranjos.
3. A festa da **Fenadoce** disponibiliza a lista das candidatas ao concurso Rainha da Fenadoce, com os dados das candidatas salvos em localStorage. Novas candidatas podem ser adicionadas pelo administrador. O campo número de votos de cada candidata deve iniciar zerado. Pode-se ainda solicitar o e-mail do votante e salvá-los em outro atributo do registro da candidata, como um vetor (para evitar que uma pessoa vote inúmeras vezes na mesma candidata).

Cronograma:

Dia 05/11:

- Entrega da descrição do trabalho, com os atributos a serem armazenados e indicação da interação e rotas.

Dia 12/11:

- Apresentação do trabalho

Conceitos:

Recursos **a**, **b** e **c** funcionando: Conceito C

Recursos **a**, **b**, **c** e **d** funcionando: Conceito B

Recursos **a**, **b**, **c**, **d** e **e** funcionando: Conceito A